Claims:

1. A method for the compression of a control traffic in media data transmission, which uses Real-time Transport Protocol (RTP) and Real-time Control Protocol (RTCP), employed particularly in real-time or near-real-time multimedia data delivery in an Internet protocol (IP) network, within the allocated fractions of the available session bandwidth, wherein the method comprises the steps of:

   initialising the context of control traffic flow by initially transmitting context parameters and

   updating said context during the session if necessary using compressed control packets.

2. The method according to claim 1, wherein the context parameters are categorised into static context parameters and dynamic context parameters.

3. The method according to one of claims 1 to 2, further comprising the step of omitting a-priori known context parameters.

4. The method according to one of claims 1 to 3, wherein said static context parameters are transmitted in at least one initialisation packet .

5. The method according to one of claims 1 to 4, wherein the dynamic context parameters are transmitted in initialisation, refresh packets or compressed control packets.

6. The method according to claim 5, wherein dynamic context parameters are further transmitted in source description (SDES) packets and BYE packets.

7. The method according to claim 5 or 6, wherein the method is employed for compressed control data transmission between a compressor and a decompressor, said method further comprising the step of:

   defining the packet format of said initialisation packets, said refresh packets and said compressed control packets and compressor and decompressor context parameters before the step of initialising.

8. The method according to claim 7, wherein the method further comprises the step of repairing and recovering out-of-sync context at said decompressor, if necessary.

9. The method according to one of claims 2 to 8, wherein in said initialisation step transmits initialisation values for said dynamic context parameters as references.

10. The method according to one of claims 2 to 9, further comprising the step of forming at least one initialisation packet from said static context parameters before transmitting them.

11. The method according to one of claims 2 to 10, further comprising the step of forming refresh packets and compressed control packets from said dynamic context parameters before transmitting them.

12. The method of claim 11, wherein the step of forming refresh packets and compressed control packets further forms source description packets and bye packets.

13. The method according to one of claims 2 to 12, further comprising the step of categorising said dynamic context parameters into occasionally changing context parameters, context parameters of random character, counter-like context parameters, frequently changing context parameters and context parameters that regularly change by a fixed delta.

14. The method of claim 13, further comprising the step of encoding said counter-like context parameters, said frequently changing context parameters and said context parameters that regularly change by a fixed delta using least-significant-bit (LSB) encoding.

15. The method according to one of claims 12 to 14, wherein the step of forming refresh packets and compressed control packets integrates said occasionally changing context parameters and said context parameters of random character in an un-encoded form into said formed packets and said counter-like context parameters, said frequently changing context parameters and said

context parameters that regularly change by a fixed delta in an encoded form into said packets.

16. The method according to one of claims 3 to 14, wherein said a-priori known context parameters comprise a control protocol version.

17. The method according to one of claims 2 to 16, wherein said static context parameters comprise:

padding flags for indicating whether the sender report packet contains an additional padding field at the end of the sender report packet being no part of the context parameters,

at least one synchronisation source (SSRC) identifier for identifying a packet-sender or a source of the media data transmission and

at least one contributing source (CSRC) identifier, identifying the at least one source that adds content to a data packet.

18. The method according to one of claims 13 to 17, wherein said occasionally changing context parameters comprise the following fields of a packet:

reception report count (RC) fields for indicating the number of report blocks in the packet,

source count (SC) fields for indicating the number of synchronisation sources (SSRC) or contributing sources (CSRC) in a source description (SDES) packet or identifying the number of synchronisation sources (SSRC) or contributing sources (CSRC) in a bye packet,

payload type (PT) fields identifying a packet type,

source description (SDES) items comprising information to describe packet-source's properties and

subtype fields in application-defined (APP) packets allowing a set of application-defined (APP) packets to be defined under one unique name.

19. The method according to claim 18, wherein said source description (SDES) items comprise:

canonical end-point identifier (CNAME) items to describe a user and domain name of a source,

user name (NAME) items to describe a common name of a source,

electronic email address (EMAIL) items to describe the email address of a source,

phone number (PHONE) items to describe a phone number of a source,

geographic user location (LOC) items to describe the geographic location of a source,

application or tool name (TOOL) items to describe a name of the source application producing the media data,

notice or status (NOTE) items for transient messages describing the status of a source and

private extension (PRIV) items to define experimental or application specific extensions.

20. The method according to one of claims 13 to 19, wherein said context parameters of random character comprise:

fraction lost fields for indicating the of number of packets lost divided by the number of packets excepted to be received and

fields (BLP) comprising a bitmask of lost packets.

21. The method according to one of claims 13 to 20, wherein said frequently changing context parameters comprise:

Real-time Transport Protocol (RTP) Timestamp field for indicating the delay since the last sender report received,

timestamp fields of the last sender report,

inter-arrival jitter fields for indicating an estimate of the statistical variance of the real-time protocol (RTP) data packet inter-arrival time and

length fields indicating the length of a packet.

22. The method according to one of claims 13 to 21, wherein said counter-like context parameters comprise:

real-time protocol (RTP) sequence numbers,

fields indicating the extended highest sequence number of received packets,

a packet-sender's packet count for indicating the total number of real-time protocol (RTP) packets a sender has transmitted in the time frame between the beginning of the session and the generation of the packet comprising the sender's packet count,

a packet-sender's octet count for indicating the total number of payload octets transmitted in real-time protocol (RTP) packets by the sender in the time frame between the beginning of the session and the generation of a packet comprising the sender's octet count and

fields for indicating the cumulated number of packets lost during transmission.

23. The method according to one of claims 11 to 22, wherein said compressed control packets can be sender report packets, receiver report packets and application-defined (APP) packets.

24. The method of claim 10 or 11, wherein said step of forming initialisation packets forms initialisation packets comprising:

a context identifier that identifies the state of the header decompressor to be used to decompress the packet,

a packet identifier to enable the packet receiver to identify the packet type,

profile information comprising the packet-sender's profile information,

a cyclic redundancy check (CRC) field for checking data integrity of the updating packet,

a static information chain comprising static context parameters and

a dynamic information chain comprising dynamic context parameters.

25. The method of claim 11, wherein said step of forming refresh packets and compressed control packets forms refresh packets comprising:

a context identifier that identifies the state of the header decompressor to be used to decompress the packet,

a packet identifier, to enable the packet receiver to identify the packet type,

profile information comprising the packet-sender's profile information,

a cyclic redundancy check (CRC) field, for checking data integrity of the updating packet and

a dynamic information chain, comprising dynamic context parameters.

26. The method of claim 11, wherein said step of forming refresh packets and compressed control packets forms sender report packets comprising a sender report packet header and at least one report block.

27. The method according to claim 26, wherein said sender report packet header comprises:

a packet identifier to identify the sender report packet type,

a reception report count field for indicating the number of report blocks comprised in the sender report packet,

an active sender flag for indicating whether a session participant who generates the report block is active or not,

a cyclic redundancy check (CRC) field, for checking data integrity of the sender report packet,

a padding flag for indicating whether the sender report packet contains an additional padding field at the end of the sender report packet being no part of the context parameters,

a least-significant-bit (LSB) encoded real-time protocol (RTP) timestamp,

an extension flag for indicating that the sender report packet further comprises an extension field,

a least-significant-bit (LSB) encoded sender's packet count field indicating the total number of real-time protocol (RTP) packets the sender has transmitted in the time frame between the beginning of the session and the generation of the sender report packet,

a least-significant-bit (LSB) encoded sender's octet count field indicating the total number of payload octets transmitted in real-time protocol (RTP) data packets by the sender in the time frame between the beginning of the session and the generation of the sender report packet, and

a length field for indicating the length of the sender report in least-significant-bit encoded format.

28. The method according to one of claims 11 to 27, wherein said step of forming refresh packets and compressed control packets forms compressed receiver report packets comprising a receiver report packet header and at least one report block.

29. The method according to claim 27, wherein said receiver report packet header comprises:

a packet identifier to identify the compressed receiver report packet type,

a reception report count field for indicating the number of report blocks comprised in the receiver report packet,

an active sender flag for indicating whether a session participant who generates the report block is active or not,

a cyclic redundancy check (CRC) field, for checking data integrity of the receiver report packet,

a padding flag for indicating whether the compressed real-time control protocol (RTCP) receiver report packet contains an additional padding field at the end

of the compressed real-time control protocol (RTCP) receiver report packet being no part of the context parameters and

a length field for indicating the length of the sender report in least-significant-bit encoded format.

30. The method according to one of claims 26 to 29, wherein said report block comprises:

a fraction lost field for indicating the of number of packets lost divided by the number of packets excepted to be received,

a least-significant-bit (LSB) encoded cumulated loss field for indicating the cumulated number of packets lost during transmission,

a least-significant-bit (LSB) encoded sequence number cycle field for indicating the sequence number cycle of the extended highest sequence number of received packets,

a least-significant-bit (LSB) encoded highest sequence number for indicating the highest sequence number received by the sender of the packet,

a least-significant-bit (LSB) encoded inter-arrival jitter field for indicating an estimate of the statistical variance of the real-time protocol (RTP) data packet inter-arrival time,

a least-significant-bit (LSB) encoded real-time protocol (RTP) timestamp and

a least-significant-bit (LSB) encoded delay since last sender report field for indicating the delay since the sender report received last.

31. The method according to one of claims 28 to 30, wherein said sender report packets and receiver report packets further comprises a field for profile-specific extensions.

32. The method according to one of claims 11 to 31, wherein the step of forming refresh packets and compressed control packets forms application-defined (APP) packets comprising:

a packet identifier to identify the application-defined (APP) packet type,

a feedback type field for indicating a feedback type comprised in the application-defined (APP) packet,

a least-significant-bit (LSB) encoded feedback length field for indicating the length of the application-defined (APP) packet and

a bitmask (BLP) field for indicating lost packets.

33. A computer program comprising program code means for executing all steps of any one of the claims 1 to 32 when said program is run on a computer.